

PowerBuilder 9.0

软件开发项目实践



软件开发项目实践

方小伟 赵永军 李其中 编著



清华大学出版社

内 容 简 介

本书通过 8 个完整的项目开发实践,系统地讲述了使用 PowerBuilder 9.0 开发数据库系统的过程和方法。全书共分 9 章,分别将这些实例按照由易到难的顺序和针对不同的应用背景来编排。将 PowerBuilder 9.0 的知识点和开发技巧分散在具体的项目开发之中,使读者在实践中加深理解,学习掌握使用大型可视化数据库开发工具、数据库的技术,以及常用的编程方法,达到举一反三学以致用目的。

本书为你搭建 PowerBuilder 编程的坚实基础,可作为大专院校相关专业学生的学习辅导教材和毕业设计指导用书,也可供实际项目开发的程序设计人员借鉴和参考。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

PowerBuilder 9.0 软件开发项目实践/方小伟,赵永军,李其中编著.

—北京:清华大学出版社, 2004

ISBN 7-302-08366-5

I. P... II. ①方...②赵...③李... III. 数据库系统—软件工具, PowerBuilder 9.0

IV. TP311.56

中国版本图书馆 CIP 数据核字(2004)第 025484 号

出 版 者: 清华大学出版社

地 址: 北京清华大学学研大厦

<http://www.tup.com.cn>

邮 编: 100084

社总机: 010-62770175

客户服务: 010-62776969

组稿编辑: 科海

文稿编辑: 安靖

封面设计: 杨月静

版式设计: 科海

印 刷 者: 北京科普瑞印刷有限责任公司

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 22.75 字数: 553 千字

版 次: 2004 年 4 月第 1 版 2004 年 4 月第 1 次印刷

书 号: ISBN 7-302-08366-5/TP·6020

印 数: 1~5000

定 价: 35.00 元

(如有印装质量问题,我社负责调换)

前 言

PowerBuilder 9.0 是目前开发客户机/服务器结构的应用系统的主要工具之一。本书在全面系统介绍 PowerBuilder 9.0 的基础知识上,通过剖析 8 个项目开发实例的设计与开发过程,讲解了 PowerBuilder 9.0 的高级技巧。

全书共分 9 章。

第 1 章主要介绍 PowerBuilder 9.0 的新特性、语言基础、数据库窗口技术、应用程序的构成和开发步骤等。

第 2 章~第 9 章通过 8 个完整的应用开发实例,系统全面地介绍了采用 PowerBuilder 9.0 开发数据库应用系统的过程和方法,其中包含了作者多年的开发经验和技巧。

本书的特点是将 PowerBuilder 的知识点分布在程序开发之中,读者在学习时应结合程序实例,深入体会 PowerBuilder 9.0 程序开发的思路与方法。

书中介绍了 PowerBuilder 9.0 应用系统开发的如下知识与技巧:

- 文件操作技术
- 数据库连接技术
- SQL Server 数据库设计方法
- FTP 协议及使用
- Windows 系统资源的应用
- PowerBuilder 9.0 界面设计

书中的应用实例均在 PowerBuilder 9.0+Windows 2000 的环境中调试通过。

书中的示例代码可以在科海网站上下载: <http://www.khp.com.cn>。

由于作者水平所限,书中难免存在错误和不足之处,敬请广大读者批评指正。作者的 Email 地址为 support@gridtek.net,或者访问 www.gridtek.net。

作者

2004 年 4 月

目 录

第1章 PB 9.0编程基础	1
1.1 PB 9.0的特性	1
1.2 PB 9.0开发环境.....	3
1.2.1 启动PB 9.0.....	3
1.2.2 PB 9.0的工具栏.....	4
1.2.3 PB 9.0工具.....	6
1.3 PB 9.0的主要画板.....	7
1.3.1 窗口画板.....	7
1.3.2 应用画板.....	9
1.3.3 数据库画板.....	11
1.3.4 数据窗口画板.....	14
1.3.5 数据管道画板.....	16
1.3.6 函数画板.....	16
1.3.7 库画板.....	17
1.3.8 菜单画板.....	17
1.3.9 工程画板.....	19
1.3.10 查询画板.....	20
1.3.11 结构画板.....	20
1.3.12 用户对象画板.....	21
1.4 PowerScript简介.....	22
1.4.1 PowerScript语言基础	22
1.4.2 数据类型.....	23
1.4.3 变量.....	25
1.4.4 常量.....	28
1.4.5 数组.....	29
1.4.6 外部函数.....	29
1.4.7 结构.....	30
1.4.8 运算符和表达式.....	31
1.4.9 运算符的优先级.....	31
1.4.10 表达式的数据类型.....	32
1.4.11 常用PowerScript语句	33
1.5 SQL语句的使用.....	38

1.5.1 嵌入式SQL语句中PowerScript 变量的调用	38
1.5.2 脚本中的错误处理	39
1.5.3 嵌入式SQL语句.....	39
1.5.4 动态SQL语句	43
1.6 常用控件	47
1.6.1 窗口控件简介	47
1.6.2 控件属性的修改	48
1.7 数据窗口对象.....	49
1.7.1 数据窗口的创建	49
1.7.2 显示风格	51
1.7.3 数据源.....	52
1.7.4 数据窗口的基本事件和 常用函数	53
1.8 小结	54
第2章 记事本应用系统	55
2.1 应用程序的创建.....	55
2.2 窗口的创建.....	58
2.3 菜单的创建.....	60
2.4 各对象脚本的编写	61
2.4.1 应用对象脚本的编写	62
2.4.2 窗口脚本的编写	62
2.4.3 菜单脚本的编写	63
2.5 应用程序的调试.....	69
2.5.1 调试环境及基本方法	69
2.5.2 断点的设置.....	71
2.5.3 调试应用程序	72
第3章 同学录管理系统	74
3.1 系统需求分析.....	74
3.1.1 系统目标设计	74
3.1.2 系统功能分析	74
3.2 数据库的创建.....	75

3.2.1 数据库文件的创建	75	5.3.3 数据库连接的建立	149
3.2.2 数据库画板的配置	76	5.4 应用程序对象的创建	150
3.3 数据库设计	78	5.5 全局变量和全局函数的定义	150
3.3.1 数据库的分析	78	5.5.1 全局变量的定义	151
3.3.2 数据表的创建	79	5.5.2 全局函数的定义	151
3.4 各对象的创建	83	5.5.3 结构的定义	154
3.4.1 应用对象的创建	83	5.6 各对象的设计及脚本编写	155
3.4.2 数据窗口对象的创建	83	5.6.1 系统维护子系统	155
3.4.3 窗口的创建	87	5.6.2 档案著录子系统	161
3.5 各对象脚本的编写	89	5.6.3 附加库管理子系统	167
3.5.1 应用对象脚本的编写	89	5.6.4 编辑库结构子系统	174
3.5.2 窗口对象脚本的编写	90	5.7 应用程序的运行	184
3.6 应用程序的编译和运行	94	5.7.1 库结构的编辑	185
3.6.1 应用程序的运行	94	5.7.2 档案的著录	186
3.6.2 应用程序的编译	94	5.8 小结	187
3.7 小结	96		
第4章 设备管理信息系统	97	第6章 项目管理系统	189
4.1 系统需求分析	97	6.1 系统设计	189
4.2 数据库设计	98	6.2 数据库设计	190
4.2.1 数据流程分析	98	6.3 数据库的实现	191
4.2.2 数据表的创建	100	6.4 应用程序对象的创建	192
4.3 应用对象的创建	104	6.5 全局函数和结构的定义	193
4.4 各功能模块的实现	105	6.5.1 全局函数的定义	193
4.4.1 设备台账管理子系统	106	6.5.2 结构的定义	196
4.4.2 租赁计划子系统	114	6.6 各对象的设计及其脚本的编写	196
4.4.3 设备租赁子系统	121	6.6.1 系统维护子系统	196
4.4.4 设备归还子系统	129	6.6.2 项目管理子系统	205
4.4.5 设备维修管理子系统	131	6.6.3 项目过程维护	210
4.5 小结	137	6.7 应用程序的运行	214
第5章 人事管理系统	139	6.7.1 系统人员信息的编辑	215
5.1 系统设计	139	6.7.2 项目信息的编辑	215
5.2 数据库设计	139	6.8 小结	216
5.2.1 基本信息表	140		
5.2.2 附件情况表	144	第7章 ftp文件传输系统	217
5.2.3 系统表	146	7.1 系统设计	217
5.3 数据库的实现	147	7.2 数据库设计	218
5.3.1 数据库的创建	147	7.2.1 系统用户信息表	218
5.3.2 数据库表单的创建	148	7.2.2 用户远程文件访问表	218
		7.3 数据库的实现	218
		7.3.1 数据库表单的创建	218

7.3.2 建立数据库连接	220	8.3 应用程序的运行	288
7.4 ftp的发布	221	8.4 小结	289
7.4.1 登录用户的创建	221	第9章 进销存管理系统	291
7.4.2 ftp文件目录的创建	222	9.1 系统设计	291
7.5 应用对象的创建	224	9.2 数据库设计	292
7.6 全局变量和全局外部函数的定义	225	9.2.1 用户基本信息表	292
7.6.1 全局变量的定义	225	9.2.2 客户设置表	293
7.6.2 全局外部函数的定义	225	9.2.3 物资库表	293
7.6.3 全局函数的定义	226	9.2.4 物资到货信息表	293
7.7 各对象的设计及其脚本的编写	227	9.2.5 库存表	294
7.7.1 系统维护	227	9.2.6 物资入库信息表	294
7.7.2 文件的上传	237	9.2.7 物资出库信息表	295
7.7.3 文件的下载	247	9.3 数据库的实现	296
7.8 应用程序的运行	255	9.3.1 数据库表单的创建	296
7.8.1 超级用户登录	256	9.3.2 数据库的连接	296
7.8.2 控件的注册	256	9.4 应用对象的创建	297
7.8.3 新用户的添加	256	9.5 全局变量和结构的定义	297
7.8.4 设置用户访问远程文件的权限	257	9.5.1 全局变量的定义	297
7.8.5 文件的上传和下载	257	9.5.2 结构的定义	298
7.9 小结	259	9.6 各对象的设计及其脚本的编写	298
第8章 数据转换程序	260	9.6.1 系统维护子系统的设计	298
8.1 实例概述	260	9.6.2 到货录入子系统的设计	323
8.2 各对象的设计及其脚本的编写	260	9.6.3 库房管理子系统的设计	336
8.2.1 应用程序对象	261	9.6.4 统计查询子系统的设计	342
8.2.2 数据窗口对象	261	9.7 应用程序的运行	355
8.2.3 应用程序窗口对象	265	9.8 小结	355

第 1 章 PB 9.0 编程基础

PowerBuilder（本书简称PB）是目前被广泛应用的一种面向对象的、具有可视化图形界面的快速交互式数据库应用开发工具，其自由的编程风格、强大的数据库处理功能、先进的Web应用技术以及友好的界面，给用户留下了深刻的印象。它可以设计传统的基于客户机/服务器体系结构的高性能应用程序，也可以用于开发基于因特网的应用系统。最新版的PowerBuilder 9.0（本书简称PB 9.0）支持应用系统同时访问多种数据库。为了开发方便，PB 9.0提供了大量的控件，既丰富了应用程序的表达能力，也加快了项目的开发进度。它还支持分布式应用系统的开发，形成多层次的应用体系结构，也就是说，它可以方便地构造应用服务器，从而形成分布式应用。

1.1 PB 9.0的特性

PB 9.0扩展了PB 8.0的功能，提供更为开放的开发环境。它包含一个优秀的代码基础和新的增强特性，以及与Web和N层开发的进一步整合。它提供了与Sybase的Enterprise Application Server（EAServer）更为紧密的集成，能够在EAServer中调用Enterprise JavaBean（EJB），并能够将现有的组件连接到EAServer。

以下简单地介绍PB 9.0的其他主要特性。

1. XML DataWindow

将DataWindow的报告数据作为扩展标记语言（XML）进行存储。用户能从DataWindow中选择数据值，然后按照用户定义的XML输出模板将其输出为XML。新的Save-As（另存为）模板视图允许用户可视化地定义新的XML输出模板，通过DataWindow定义生成默认的模板，或者输入现有的XML模板。

XML输出模板由能够全面定制的元素和属性名称组成，它们将被映射到特定的DataWindow列、计算字段和其他数据值。表达式或格式符号可以用于数据数值，以便为不同的显示格式生成XML数值。

每个XML输出模板都将按名称存储，用户能为每个DataWindow创建多个XML输出模板。新的另存为XML的配置选项让用户选择在运行时使用哪个XML模板，从而允许一个DataWindow生成多个XML数据文件，每个都有自己独特的格式。同样，新的ImportFile模板能被定义用于从XML文档向DataWindow输入数据。

2. PB文档对象模型

PB文档对象模型（PBDOM）是一个完整的应用程序接口（API），它用于创建、读写和

操作XML文档。PBDOM也使通过标准的PowerScript语句可以轻松地访问和操作XML文档。PBDOM可以基于一系列非可视化类集合和使用PB本地接口技术（PowerBuilder Native Interface, PBNI）创建。

3. RAD JavaServer页面

Java Server Page (JSP) 的开发是PB 9.0的一个重要特性，它为用户提供了一个用于JSP的快速应用开发环境（RAD），同时还为高级用户提供全面的配置灵活性。PB JSP向导能帮助用户快速开发出一个JSP Web的应用程序框架，还能指导用户创建JSP Web应用实体和构建JSP页面的框架。

因为JSP页面是通用的HTML页面的扩展，在PB中编辑JSP页面就像现在编辑HTML Web页面一样。PB JSP基于现有的Web Target的实现方式来构建，这样会使用户在创建和编辑JSP页面时，感觉和处理HTML页面一样。

PB 8.0中的Web Target功能拥有一个强大的对象模型，该模型能处理复杂的数据传输、HTML的生成和用于服务器端的JavaScript的生成。PB 9.0 JSP对象模型包括PB 8.0中对象模型的所有功能，同时增加了一些针对JSP开发的功能。

在部署JSP Web应用程序的时候，PB 9.0提供了两个JSP Server供选择：EAServer 4.0和Apache的Tomcat Server。PB对JSP和Tomcat Server的支持显示了Sybase一贯承诺的对开放式平台的支持。

4. 手持设备的开发

PB的桌面应用程序可以被部署到多种手持设备上，它为使用WinCE和Palm OS的手持设备提供了支持。这二者相比，WinCE尤其受到重视。

对桌面集成驱动电子设备进行修改，使应用程序能够部署到一系列手持设备上。Project Painter是进行这些修改的场所，开发人员可以在此设定一些参数，以便通过相关的PB对象构建出一个应用程序。某个手持设备被选中后，应用程序将检测出它不支持的功能并发出相应的警告，使开发人员能够改变相应的项目或合理忽略某些问题。这样，相应的桌面PB应用程序将被转化成能够在手持设备上运行的程序。系统将为手持数据库提供DataWindow检索和更新方面的支持。

5. 高速开放式集成开发环境

PowerScript++类就是能够扩展PBS功能的C++ DLL，它们由Sybase、第三方企业或用户开发而成。通过注册器功能，PowerScript类向PB虚拟机声明自己的方法和事件。一经注册，该类能够像通常的PB对象那样使用。PowerScript++类可以采用非可见的用户对象或一个窗口形式。Callback功能能够在PowerScript中实现。另外，PowerScript提供了创建XML解析器等可重用组件的绝好方法。

6. 客户端EJB服务器支持

提供了调用第三方应用服务器中EJB的桥接技术和详尽的样本程序，使PB应用程序能访问到部署在第三方J2EE应用服务器中的EJB商业逻辑。

Sybase已经开发出一个全自动的工具来生成这些代理组件，以获得更高的生成力。这个解决方案的代号为EAServer Integrators (EASI)。

在最初的版本中，EASI允许将PB作为BEA WebLogic、IBM WebSphere或任意J2EE认证的应用服务器的客户端，PB通过一个EAServer生成并部署的proxy stateless session Bean与这些服务器连接。

7. 支持Microsoft .NET

PB 9.0支持.NET框架，通过Web Services这样一个首要措施使.NET与PB进行集成。

8. 源代码控制

为了源代码控制 (Source Code Control, SCC) 功能的显著提高，PB 9.0引入了初始化工作空间的执行线程，这样就消除了大型应用的启动延迟。同时，为修订数据的引入提供了巨大的改进——可以决定同步状态，支持新的集成扩展。

9. OrcaScript

OrcaScript允许编写批处理语句来处理PB开发环境之外的应用和文件。OrcaScript能够从源代码控制中获得最新的PowerScript对象创建PBL对象，给EAServer配置组件，编译PB可执行文件——所有这些都不需要操作人员干涉。

10. PDF支持

PB提供了两种方法将数据窗口对象保存为PDF（可移植文档格式）的报表。默认为提取方法，它能够保存数据窗口对象的所有类型。同时，数据窗口数据和显示格式也可以保存为文档或字符串。

1.2 PB 9.0开发环境

PB是一种在Windows环境下运行的应用程序开发工具。它采用面向对象的程序设计方法，加上可视化的开发界面，能够让程序员轻松地开发出相互独立的应用程序。PB的优点还体现在数据库应用开发上，其中，数据窗口控件的强大功能是所有其他开发工具无法比拟的。

1.2.1 启动PB 9.0

在Windows的“程序”菜单中选择Sybase | PowerBuilder 9.0 | PowerBuild 9.0菜单项，便可启动PowerBuilder 9.0。在Sybase PowerBuilder 9.0的启动画面后，便出现如图1-1所示的PowerBuilder 9.0的集成开发环境，它是一个多文档界面 (MDI) 窗口，也是完成应用程序开发的主要工作区。

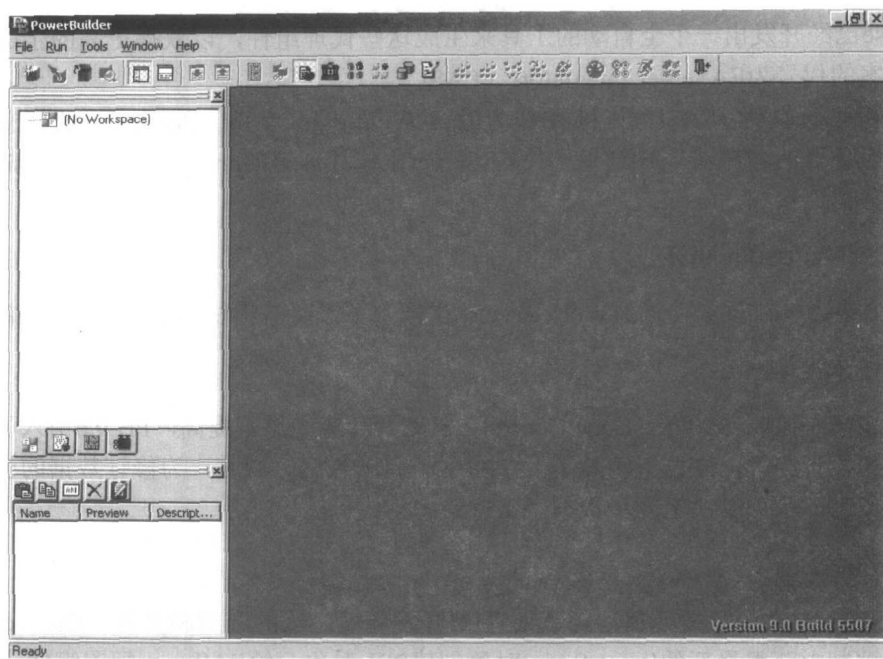


图 1-1 PB 9.0 集成开发环境

1.2.2 PB 9.0的工具栏

PB 9.0提供了2组工具栏 (Toolbar)，分别为：PowerBar (工具栏) 和PainterBar (画板工具栏)，每个工具栏都提供了多个图标按钮。

PowerBar工具栏是PB开发环境里的主要工具栏，如图1-2所示。通过PowerBar工具栏可以完成很多操作，例如，单击工具栏上的相应图标，可以生成新的对象和应用程序，也可以打开存在的对象或调试运行应用程序等。



图1-2 PowerBar工具栏

在PB中，用户可以右击工具栏，在弹出的菜单中选择Show Text，则工具条上将显示带文本的图标；也可以将鼠标指针放置在工具栏的图标上，稍等片刻，将会显示该图标的提示标签，这就是PB的PowerTip。表1-1按照从左到右的顺序列出了图标的功能。

表1-1 工具栏的功能

工具栏图标名称	功能
 New	创建新对象
 Inherit	继承原有的对象生成新对象
 Open	打开存在的对象

(续表)

工具栏图标名称	功能
Run/Preview	运行窗口或预览窗口
System tree	打开或关闭系统树窗口
Output window	打开或关闭输出窗口
Next error/Previous error	输出窗口的导航
To-Do List	跟踪应用程序的运行任务, 通过连接迅速定位到需要完成的任务
Browser	提供浏览对象或应用对象
Clip window	存储用户常使用的对象和代码
Library	打开库文件管理画板
DB profile	定义和设置连接特定数据库的参数
EAServer profile	定义和设置连接特定服务器的参数
Database	打开数据库画板, 用来维护数据库和数据表
Edit	打开编辑器编辑文本文件
Incremental Build Workspace	更新最近改变的工作空间的所有目标和对象
Full Build Workspace	更新工作空间的所有目标和对象
Delpoy Workspace	发布工作空间内所有的目标
Skip/Stop	中断所有的建造、发布和搜索操作
Debug	调试目标, 可以在代码中设置断点、查看表达式, 检查变量在执行中的变化情况
Select & Debug	选择目标和打开调试器
run	运行目标
Select & Run	选择目标且运行
Exit	关闭PowerBuilder

通过PainterBar工具栏, PB各画板完成当前画板中的任务。它在相应的画板中显示(不同的画板会有所不同), 部分画板有多个PainterBar工具栏。

用户可以根据自己的习惯设置工具栏。例如, 用户可以移动工具条的位置, 添加常用的图标或在工具条上显示文本等。右击工具栏, 可以在弹出的快捷菜单中选择需要的菜单项, 例如, 选择Floating将工具栏设置成浮动的, 如图1-3所示。



图 1-3 定制工具栏

如果要在工具栏中定制自己需要的工具, 可右击相应的工具栏, 在弹出菜单中选择

Customize, 将弹出Customize对话框, 如图1-4所示。只要将需要的工具图标从Selected palette列表框拖到Current toolbar列表框的合适位置即可。例如, 在默认的情况下, 文件编辑器图标不显示在工具栏中, 如果需要, 可以用上述方法将其添加到PowerBar工具栏上。

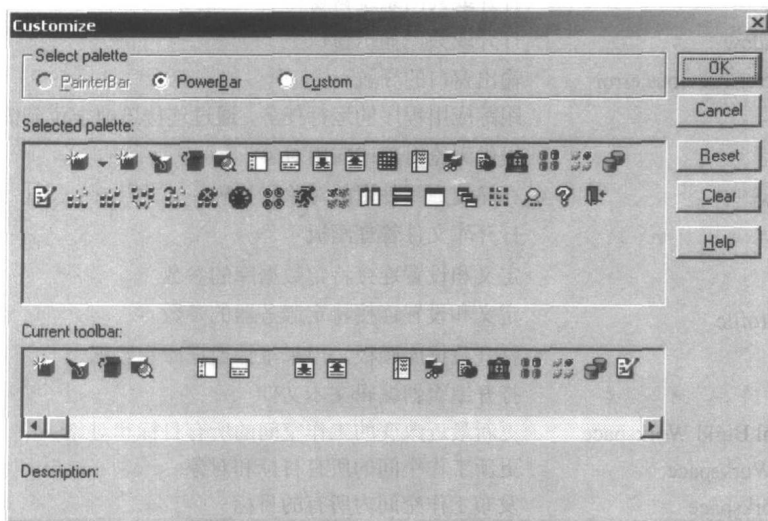


图 1-4 Customize 对话框

1.2.3 PB 9.0工具

PB提供了许多工具, 用于帮助用户进行程序开发。这些工具有的直接使用, 有的需要通过操作来打开。

在PowerBar工具栏上直接可以使用的工具有DB profile (数据库配置文件)、Browser、File Editor、Debugger和To Do List。

单击在PowerBar工具栏上的New图标, 在如图1-5所示的New对话框中选择Tool选项卡中的工具, 除了Library Painter、File Editor Migration Assistant和Web DataWindow JavaScript外, 还有以下几个工具:

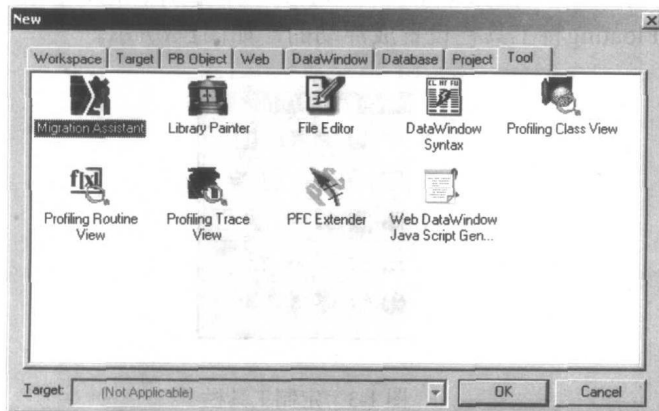


图 1-5 选择工具

- **DataWindow Syntax** 用于快速生成Describe、Modify和SyntaxFromSQL等修改数据窗口的语句。
- **Profiling Class View、Profiling Routine View、Profiling Trace View** 使用跟踪信息创建应用程序的配置文件。
- **PFC Extender** 在两个已有级别（如PFC祖先和扩展级别）之间自动由应用库创建和组装一个中间扩展级别。

1.3 PB 9.0的主要画板

PB有多个画板，用户通过这些画板来编辑应用程序的各个对象，以完成整个应用程序的开发任务，所以熟练使用各画板是使用PB开发应用程序的基础。下面将介绍各个画板的功能和使用方法。

1.3.1 窗口画板

和大多数可视化编程工具一样，用户可以在PB窗口中拖放可视化控件，设置窗口大小、类型，设置各种可视化控件的属性、样式等，并且可以在窗口的设计开发过程中预览窗口，这样可以使创建窗体更加直观、形象和快捷。此外，在窗口画板中可以直接编写窗口各对象的脚本。

用户可以通过新建窗口或打开已有窗口来打开窗口画板，下面分别介绍。

1. 通过新建窗口打开窗口画板

从PowerBar工具栏上选择New图标，在New对话框中的PB Object选项卡中选择Window图标，再单击OK按钮，或双击该图标，如图1-6所示，就可以打开窗口画板。如图1-7所示，PB在Layout视图中显示了一个空白的新窗体，然后用户就可以设置窗口属性，添加控件，添加不可见对象，以及为窗口和控件编写脚本等。



图 1-6 New 对话框中的 PB Object 选项卡

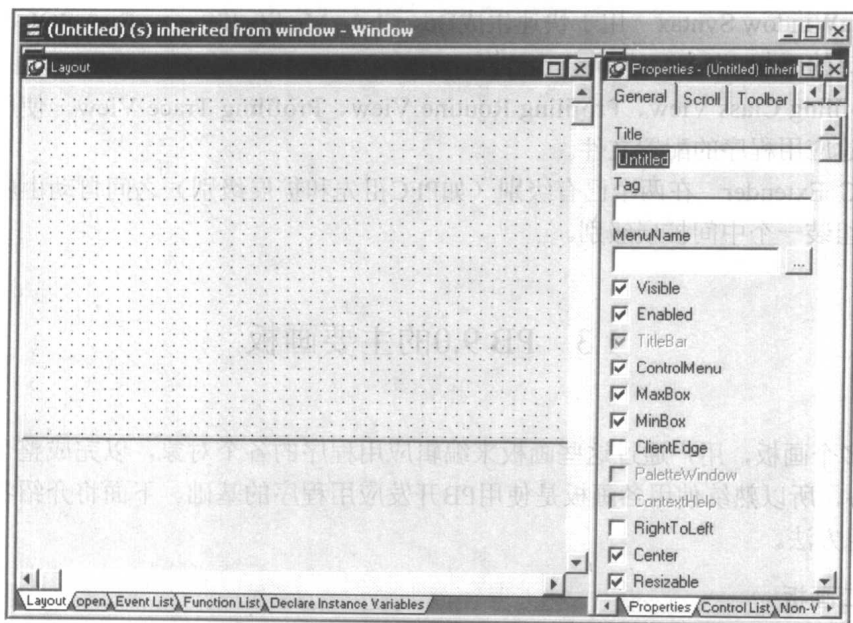


图 1-7 窗口画板样式

2. 通过打开已有窗口打开窗口画板

如果通过打开已存在的窗口来打开窗口画板，那么打开方式有两种：

一种方法是单击工具栏中的Open图标，然后从打开的对话框中选择需要打开的窗口对象。如图1-8所示，在这个对话框的Object of Type下拉列表框中选择Windows类型，可以看到，在Target下拉列表框中列出了当前工作空间的所有目标，Libraries列表框中列出了当前应用程序的所有库文件，选择保存应用程序时的库文件，此时在Object列表框中就可以显示库文件中所包含的窗口对象。选择所要打开的对象，然后单击OK按钮就可打开窗口画板。



图 1-8 Open 对话框

另一种方法是在系统树中直接打开已经存在的窗口对象。系统树窗口内包含了已经存在的窗口对象，可以双击窗口对象打开窗口画板，或者右击该窗口对象，在弹出的快捷菜单中选择Edit，同样可以打开窗口画板。

3. 画板工具栏

窗口画板也有自己的工具栏，如图1-9所示。窗口画板工具栏可以完成对窗口的各种操作。



图 1-9 窗口画板的工具栏

1.3.2 应用画板

应用画板是用来创建和编辑应用对象的，可以对应用程序的open、close、error等事件编程和对对象的属性进行定制和修改。该画板看上去很像非可视化用户对象画板，因此，应用对象可以认为是非可视化用户对象。应用画板的样式如图1-10所示。

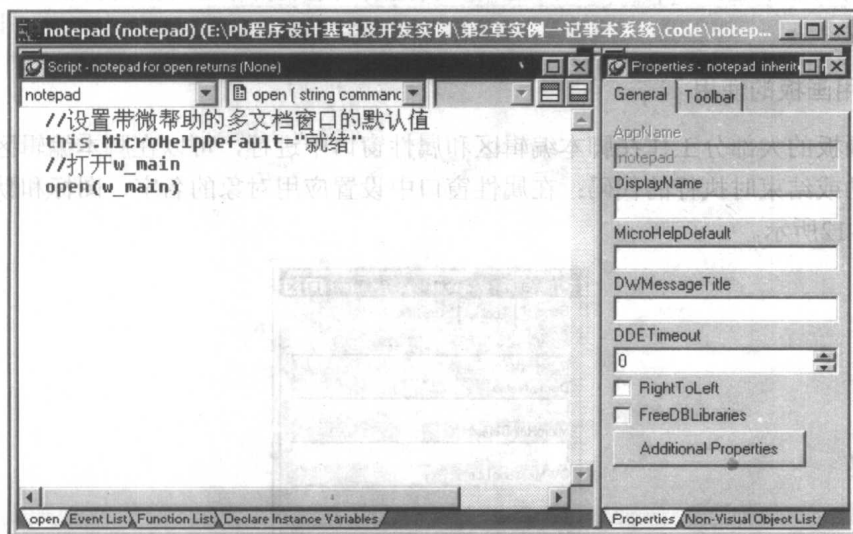


图 1-10 应用画板

在PB中，应用对象只是应用程序的一个对象，而应用程序是一个具有特定功能的应用软件。应用对象是应用程序的入口模块。

1. 应用画板的打开

可以通过新建应用对象或打开已存在的应用对象来打开应用画板。

(1) 通过新建应用对象打开应用画板

单击工具栏中的New图标，然后从弹出的New对话框中选择Target选项卡中的App-lication图标即可，如图1-11所示。

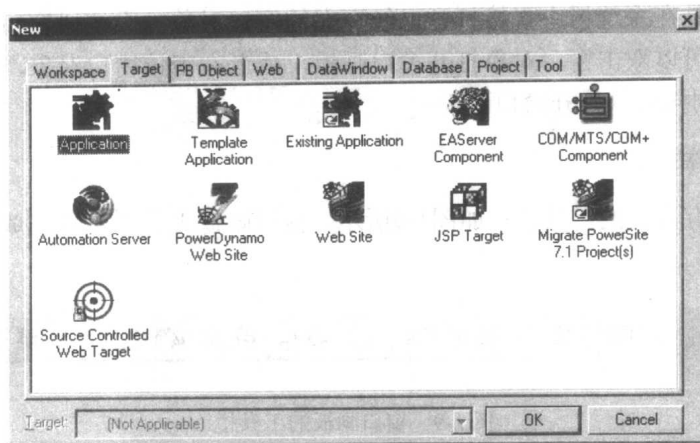


图 1-11 通过 New 对话框新建 Application 对象打开应用画板

(2) 通过打开已存在的应用对象打开应用画板

单击工具栏中的 Open 图标，在弹出的对话框中选择需要打开的应用对象（在 Object of Type 下拉列表框中选择 Application 类型），或直接从系统树中选择要打开的应用对象来打开应用画板。

2. 应用画板的使用

应用画板的大部分工作在脚本编辑区和属性窗口中进行，可以在脚本编辑区内编写应用对象启动或结束时执行的代码；在属性窗口中设置应用对象的名字、图标和默认的字体等，如图 1-12 所示。

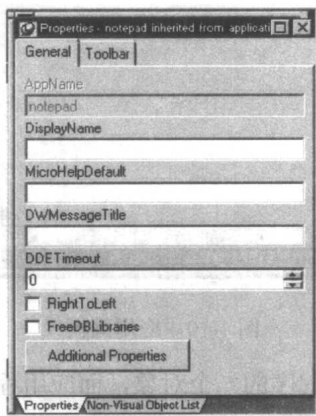


图 1-12 应用程序属性窗口

单击应用程序属性窗口上的 Additional Properties 按钮，可以进一步设置应用程序的其他属性，如图 1-13 所示。可以通过它设置开发过程中窗口、控件，以及数据窗口中的标题、内容、背景和控件的颜色、字体等属性。创建一个新应用程序时，最好设置 Additional Properties 中的属性，这样可以使开发的程序具有统一的样式，避免每次编辑时修改颜色、字体等。

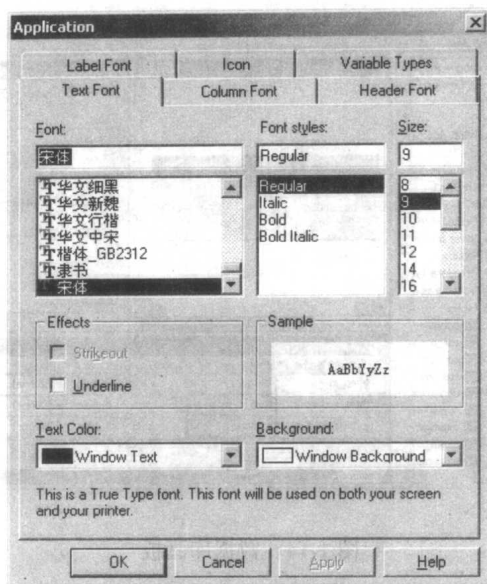


图 1-13 应用对象属性的对话框

1.3.3 数据库画板

PB开发环境提供了数据库画板，用于方便地对数据库进行管理，与数据库相关的任务绝大部分可以在数据库画板中完成。在数据库画板中，既可以创建数据表、视图、存储过程、触发器等数据库对象，也可以检索数据和执行SQL语句，还可以修改、添加、删除数据。PB通过数据库画板实现了用统一的界面来处理不同的数据库。在程序的开发过程中，数据窗口的设计、SQL语言的编译等，需要PB与相应的后台数据库连接，所以在开始编写代码前必须通过数据库画板正确连接到数据库。

单击工具栏的Database图标，打开数据库画板，如图1-14所示。在打开数据库画板前，PB已经和当前数据库进行了连接。第一次运行PB打开数据库画板时，当前数据库是安装PB时同时安装的示例数据库（EAS Demo DB V9）。如果不是第一次打开数据库画板，那么数据库画板将打开上一次连接的数据库。要在数据库画板上连接数据库，必须配置Profile，下面先来介绍Profile的配置。

Profile用来配置数据库的连接参数，这些参数将存储于系统的注册表中。在PB开发环境中，在Profile的配置之前，要保证数据源配置正确，这是因为Profile通过已存在的数据源连接数据库，如果数据源配置不正确，就无法正常连接数据库。一般的数据体可以通过ODBC（面向对象数据库连接）通用数据接口来连接，而Microsoft SQL Server和Oracle等大型数据库可以通过其专用接口来连接。

ODBC的配置有两种方法：一种方法是使用系统的控制面板中的ODBC Data Sources实现；另一种方法是ODBC下的Utilities中的ODBC Administrator选项，如图1-15所示，此时会打开ODBC配置对话框，如图1-16所示。ODBC的具体配置这里就不再赘述。